

JUN 13 2005

**PRINTER RUSH**  
(PTO ASSISTANCE)

Application : 09675066 Examiner : Chan GAU : 2183  
From : ewc Location : IDC FMF FDC Date : 6-06-05

Tracking # : epm 09675066 Week Date : 4-25-05

ATTN CHIEF DRAFTSPERSON

DOC CODE	DOC DATE	MISCELLANEOUS
<input type="checkbox"/> 1449		<input type="checkbox"/> Continuing Data
<input type="checkbox"/> IDS		<input type="checkbox"/> Foreign Priority
<input type="checkbox"/> CLM		<input type="checkbox"/> Document Legibility
<input type="checkbox"/> IIFW		<input type="checkbox"/> Fees
<input type="checkbox"/> SRFW		<input type="checkbox"/> Other
<input checked="" type="checkbox"/> DRW	<u>2-06-04</u>	
<input type="checkbox"/> OATH		
<input type="checkbox"/> 312		
<input type="checkbox"/> SPEC		

[RUSH] MESSAGE:

There are red-lined corrections in the figures  
presented on 2-06-04

Thank you

[XRUSH] RESPONSE:

"print as is"

INITIALS: EBR

NOTE: This form will be included as part of the official USPTO record, with the Response document coded as XRUSH.

REV 10/04

00675055-002800

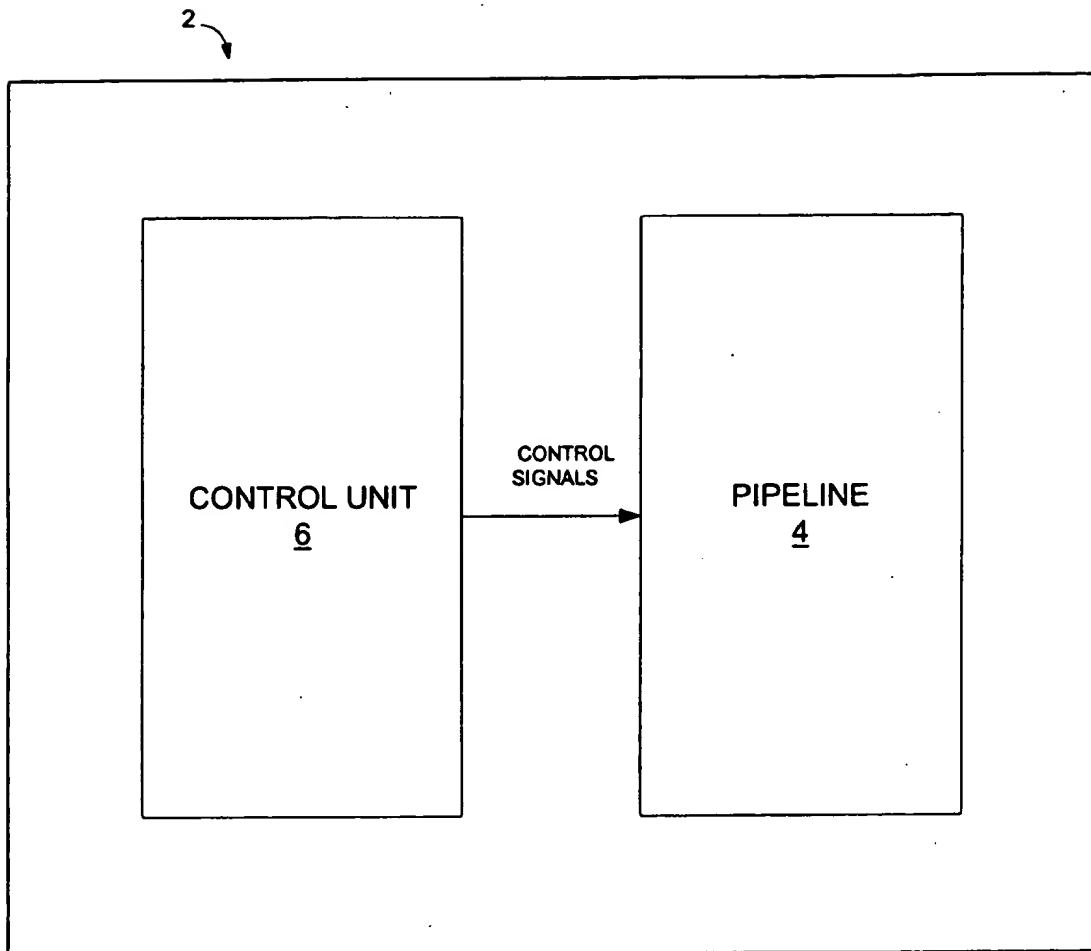


FIG. 1

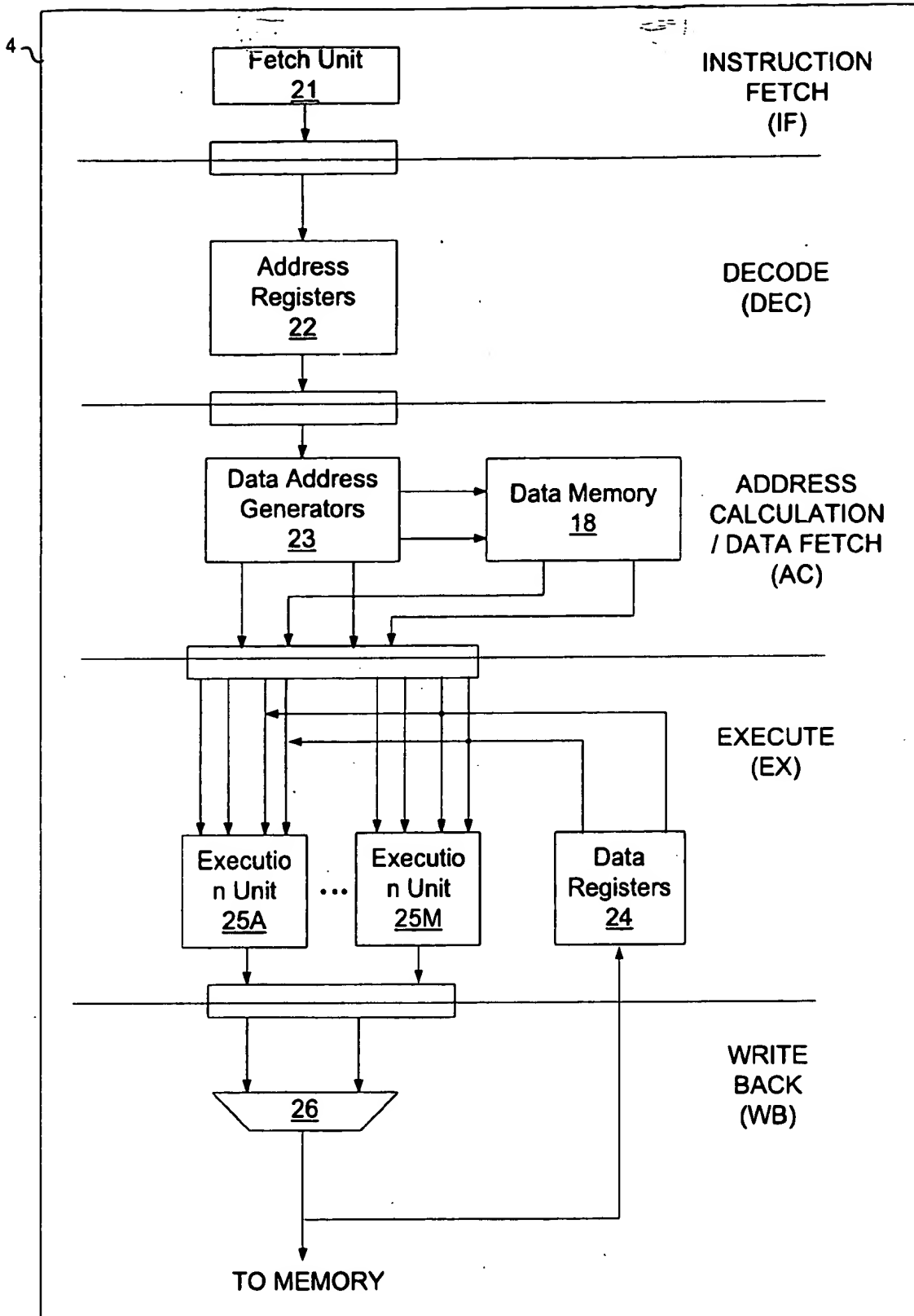


FIG. 2

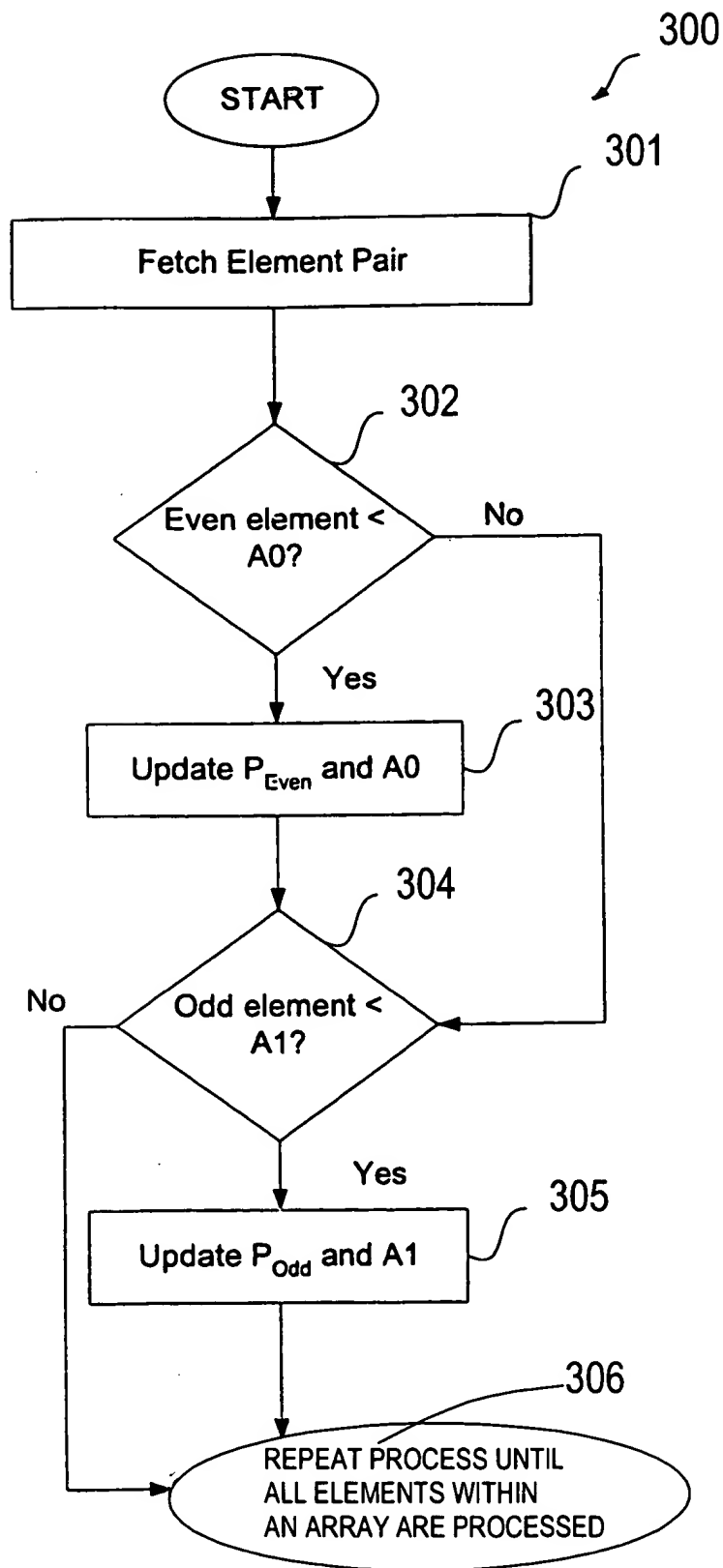


FIG. 3

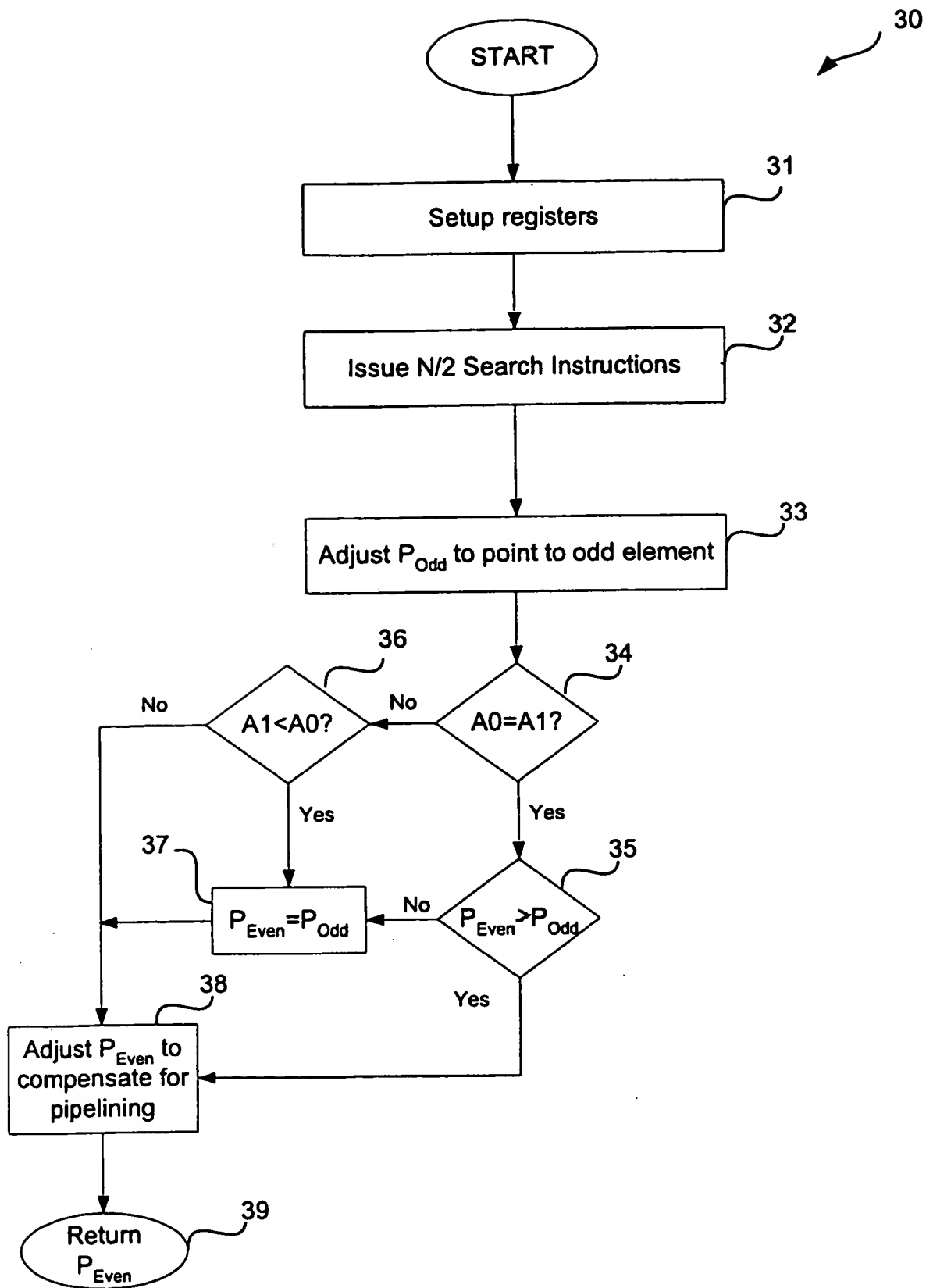
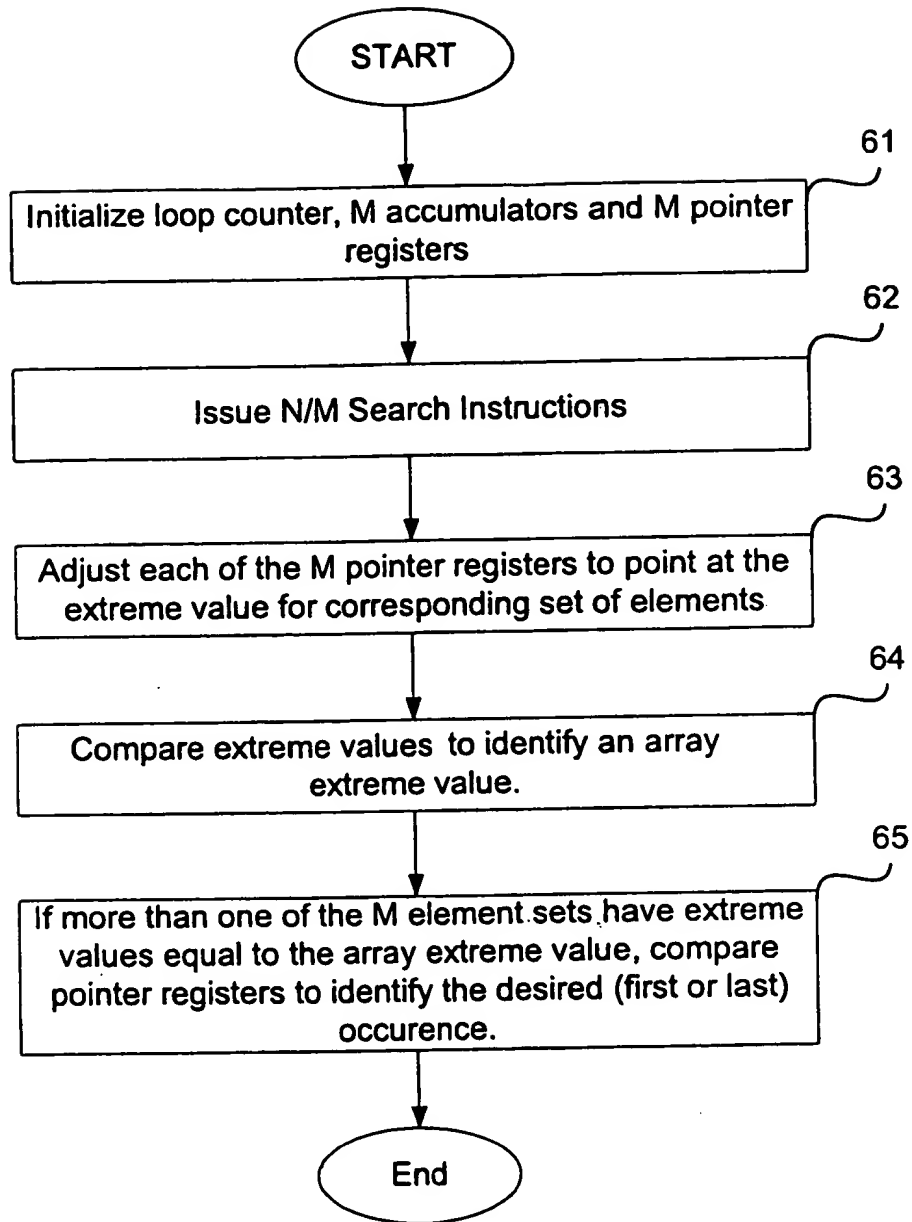


FIG. 4



**FIG. 6**

```
graph TD; START([START]) --> 61[Initialize loop counter, M accumulators and M pointer registers]; 61 --> 62[Issue N/M Search Instructions]; 62 --> 63[Adjust each of the M pointer registers to point at the extreme value for corresponding set of elements]; 63 --> 64[Compare extreme values for to identify an array extreme value.]; 64 --> 65[If more than one of the M element sets have extreme values equal to the array extreme value, compare pointer registers to identify the desired (first or last) occurrence.]; 65 --> END([End]);
```

The flowchart illustrates the N/M Search process. It begins with a **START** terminal, leading to step 61: **Initialize loop counter, M accumulators and M pointer registers**. This is followed by step 62: **Issue N/M Search Instructions**. Step 63: **Adjust each of the M pointer registers to point at the extreme value for corresponding set of elements**. Step 64: **Compare extreme values for to identify an array extreme value.** Step 65: **If more than one of the M element sets have extreme values equal to the array extreme value, compare pointer registers to identify the desired (first or last) occurrence.** The process concludes at the **End** terminal.

FIG. 6